支持增量学习的文本单类别分类算法

戴 洪[1],朱 明[2],刘守群[2]
DAI Hong[1],ZHU Ming[2],LIU Shou-qun[2]

1.中国科学技术大学 电子信息工程系,合肥 200027
2.中国科学技术大学 自动化系,合肥 200027
1.Department of Electronic Engineering and Information Science,University of Science & Technology of China,Hefei 230027,China
2.Department of Automation,University of Science & Technology of China,Hefei 230027,China
E-mail:daihong@mail.ustc.edu.cn

DAI Hong,ZHU Ming,LIU Shou-qun.Incremental learning algorithm for one-class document classification.Computer Engineering and Applications,2008,44(27):157-158.

Abstract: In this paper,an incremental learning algorithm for one-class document classification is proposed.It also has the advantage of low computational load with the same level of performance.A prototype system is constructed to implement the algorithms,and the results of the tests are pretty good.

Key words: Naïve Bayesian;Support Vector Machine(SVM);one-class classification;text/Web classification

摘 要: 目前的文本单类别分类算法在进行增量学习时需要进行大量的重复计算,提出了一种新的用于文本的单类别分类算法,在不降低分类效果的同时,有效地减少了加入新样本学习时所需的计算量,从而比较适合于需要进行增量学习的情况。该方法已进行了测试实验,获得了较好的实验结果。

关键词: 简单贝叶斯;支持向量机;单类别分类;文本/网页分类
DOI: 10.3778/j.issn.1002-8331.2008.27.050 **文章编号:** 1002-8331(2008)27-0157-02 **文献标识码:** A **中图分类号:** TP181

1 引言

传统的分类学习算法通常基于正例和反例的数据,当反例数据不容易获得或者数量太多使得有限的反例数据不足以表现反例集特征的时候,这种学习方法就会遇到困难。单类别分类(one-class classification)算法仅利用正例样本来进行分类学习,较好地解决了上述问题。近年来,各种传统的算法都被修改以适应单类别分类,包括最近邻方法[1],简单贝叶斯(NB)[2-3],神经网络[2],支持向量机(SVM)[1,5]等。单分类算法已被广泛应用在特定主题网页分类、垃圾邮件鉴别,甚至恶意代码的检测上。

但是已有方法在进行增量学些时需要大量的重复计算,往往一个新样本的加入就造成所有数据失效,从而需要全面的重复计算。针对以上单分类算法存在的问题,本文提出了一种新的单类别文本分类算法(One class Document Classification ILODC),该方法较好在不降低分类效果的同时,有效地减少了加入新样本学习时所需的计算量,从而比较适合于需要进行增量学习的情况。

2 ILODC 算法概述

ILODC 算法的基本思想是同类文本中,总应该出现一些关键性的词汇,如果某一文本中这些词汇出现的概率太低,或是根本没有这些词汇出现,则认为不是属于该类的。同时为了支持增量学习,ILODC 通过在训练样本集中选取这些关键性词汇来体现整体性质,而在计算每篇文档的特征时则只和关键词汇和该文档本身的性质相关, 而与训练样本集合的其他性质无关,从而使得新样本加入时所需要的重复计算大量减少。

ILODC 算法步骤:

(1)根据文档出现频率(document-frequency[2],指样本集中包含这个词的样本的数目)排好序,选取最高的 m 个词作为关键词集合 S_{key}。

(2)对于每一个样本 d,计算频率特征向量(frequency representation):

$(f_1,f_2,\cdots,f_m)^T$

f_i 用下式表示:

$$f_i=\frac{n_{wi}+\frac{n}{E_N}}{n+m\frac{n}{E_N}} \quad (1)$$

其中 n_{wi} 是第 i 个关键词 wi 在样本 d 中出现的次数,n 是样本 d 中的词语总数,m 是关键词的总数,E_N 是所有 N 篇训练样本词数的均值。

(3)计算 $f(d)$

基金项目: 国家高技术研究发展计划(863)(the National High-Tech Research and Development Plan of China under Grant No.2006AA01Z449)。
作者简介: 戴洪(1982-),硕士生,主要研究方向:文本分类、信息安全;朱明(1964-),教授,博士生导师,主要研究方向:分布式智能服务系统、多媒体数据挖掘、网络安全;刘守群(1982-),博士生,主要研究方向:信息安全、网页信息抽取、搜索技术。
收稿日期: 2007-11-09 **修回日期:** 2008-01-31

$$f(d)=\prod_{i=1}^{m}f_i=\prod_{i=1}^{m}\frac{n_{ui}+\frac{n}{E_N}}{n+m\frac{n}{E_N}} \quad (2)$$

(4)类别判断

设 $Threshold=\min\{f(d_i)|i\in[1,N],d_i\in \text{training doucuments}\}$ (3)

则对于某一测试样本 d_x,计算出 $f(d_x)$,如果 $f(d_x)<Threshold$ 则判定 d_x 不属于训练样本的类别。

因此 ILODC 认为当测试样本特征 $f(d)$比训练样本中最小的 $f(d)$还小时,则认为它和训练样本不是同类别。

3 ILODC 增量学习

为了支持增量学习,就要在不影响分类效果的同时尽可能在单个样本的计算时较少涉及到整体的性质。当前的分类算法当训练集发生改变时,就要进行大量的重复计算。例如:NB 算法[2]有新的训练样本加入后,需要重新计算所有的 $p(w|E)$,即每一个 $p(d|E)$需要重新计算;SVM[1]算法在有新的训练样本加入时,需要重新计算超平面,这是一个最优化问题,最终不可避免的要进行大量的矩阵运算,即要求向量 $\boldsymbol{w}$ 和标量 ρ[1]使得 $\min\frac{1}{2}\|\boldsymbol{w}\|^2+\frac{1}{v1}\sum_{i=1}^{1}\varepsilon_i-\rho$,同时满足$(w\cdot\Phi(x_i))\geqslant\rho-\varepsilon_i$。

ILODC 算法考虑的是关键词在每一个样本 d 中的分布是否满足是正例样本的最低要求。当有新的训练样本加入时,首先更新关键词的集合 S_{key}。如果没有出现新的关键词代替旧的关键词或者原有的关键词只是在顺序上发生了变化,都无需对以前已经计算好的 $f(d)$进行重复的计算。在需要重新计算 $f(d)$时,只需计算新加入关键词的 f 值,其它已经计算并仍然保留在 S_{key} 中的关键词的 f 值是可以复用的。

下面试以一例说明:假设一开始训练文本的 $S_{key}=\{red,blue\}$,当有新的样本加入时,会出现两种情况:

(1)S_{key} 仍然是$\{red,blue\}$。由公式(2)可知这时所有老训练样本的 $f(d)$都没有变化,只要计算新训练样本的 $f(d)$值,再将它们与已有的 Threshold 比较就可以得出新的 Threshold。

(2)S_{key} 发生了变化,假设变为$\{red,green\}$,即 *green* 取代了 *blue* 成为新的关键词。这时需要对每个样本 $f(d)$进行重新计算,但也无需重头开始,那些没有改变的关键词 f_i 仍然可用,新的特征:$f(d)_{new}=\frac{f(d)_{old}\prod_{uj\in S_{add}}f_{uj}}{\prod_{ui\in S_{del}}f_{ui}}$。$S_{add}$ 为新加关键词集合,S_{del} 为被替换的关键词集合,对于上面的例子 $f(d)_{neu}=\frac{f(d)_{old}f_{green}}{f_{blue}}$。这在训练数据比较稳定,只有少量关键词出现替换时尤其有用。

最后补充一句上以上分析建立在新的样本加入时训练样本词数均值 E_N 变化不大的基础上。事实上当训练样本比较稳定时 E_N 的变化确实很小。如果新的样本使得 E_N 变化很大,则需要重新计算所有样本的 $f(d)$。

可见 ILODC 选择了训练集合的两个比较稳定的性质 S_{key} 和 E_N 作为学习时用到的整体的特征,只要它们保持相对的稳定,ILODC 在进行增量学习时就只需要很少的计算,而已有算法则无法做到这一点。

同时 ILODC 具有良好的映射性质。从式(2)可知,所有的没有任何关键词出现的样本都有相同的 $f(d)$:

$$f(d)=f_{origin}=\left(\frac{1}{E_N+m}\right)^m$$

这样既避免了某些正例样本因为缺少少量关键词而被错误地分在反例类别中,又避免了那些没有关键词或只有极少数关键词出现的反例样本因为文本短而被错误地分在正例类别中。

由公式(3),可得 Threshold 是训练样本中与没有任何关键词出现的样本(潜在的反例样本)最近的样本。顺便提一下,也可以取 $Threshold_{neu}=\frac{Threshold+f_{origin}}{2}$,这时 ILODC 算法也成为了一个 SVM 算法的特例。

4 实验及说明

为了说明 ILODC 有与已有算法相当的分类效果,进行了测试,实验工作平台为 Windows 操作系统,使用的语言是 Java,并用到了开源的 Java 中文分词项目和英文的停用词表。

实验结果统一以 F 参数进行衡量,F 可以由召回率(Recall)和精确率(Precision)计算得出:

$$R=\frac{TP}{TP+FN};P=\frac{TP}{TP+FP}$$

$$F=\frac{2RP}{R+P}=\frac{2TP}{2TP+FP+FN}$$

其中 TP 为测试结果中正确肯定的样本数目,FN 为错误否定的样本数目,TN 为正确否定的样本数目,FP 为错误肯定的样本数目。

第一组实验选取了标准测试数据集 Reuters-21 578[7]。Reuters 数据集由 135 个类别的共计 21 578 篇短文组成。按照数据集中的 ModApte 分类法,选取了 2 个小类进行了实验。表 1 是实验数据的规模,表 2 是实验和与文献[2]的结果比较,这里统一关键词个数 m=50。

表 1 ModApte 的测试数据规模

类别	训练样本	正例测试样本	反例测试样本
earn	2 698	1 040	267
grain	334	116	267

表 2 ModApte 测试结果 F 值比较

	神经网络	支持向量机	ILODC
earn	0.932	0.807	0.888
grain	0.476	0.317	0.517

测试 ILODC 之前的预处理包括统一大小写和去除停用词,而文献[2]中的除了上述两项外还有更加细致的处理步骤。

第二组实验是为了测试在中文条件下 ILODC 的表现。由于没有标准的数据集,手工抽取了两个类别的网页文本内容,一组是 71 个和 2007 年中国企业 500 强有关的新闻网页,其中的 50 个页面用作训练样本,剩下的 21 个作正例测试样本;另一组是用来做反例测试样本的 32 个有关中国队 49 届世乒赛的新闻网页。

中文的实验结果列在表 3 中并与本文实现的单类别 NB[2,5] 分类算法比较。

表 3 中文的测试结果

	TP	*FN*	*TN*	*FP*	*F*
ILODC	18	3	31	1	0.90
NB	21	0	8	24	0.64

由于 NB 算法有不同的实现,现将实现的 NB 算法简单描述如下:

(下转 164 页)